



Coupling technologies for Earth System Modelling

S. Valcke¹, V. Balaji², A. Craig³, C. DeLuca⁴, R. Dunlap⁵, R. W. Ford⁶, R. Jacob⁷, J. Larson⁸, R. O’Kuinghttons⁴, G. D. Riley⁹, and M. Vertenstein³

¹CERFACS, Sciences de l’Univers au CERFACS, URA1875, 42 Av. G. Coriolis, 31057 Toulouse Cedex 01, France

²Princeton University, Princeton, NJ, 08544, USA

³National Center for Atmospheric Research, 1850 Table Mesa Drive, Boulder, CO, 80305, USA

⁴NOAA/CIRES 325 Broadway, Boulder, CO, USA

⁵College of Computing, Georgia Tech, 266 Ferst Drive, Atlanta, GA, USA

⁶STFC Daresbury Laboratory, Warrington, WA4 4AD, UK

⁷Mathematics and Computer Science Division, Argonne National Laboratory, 9700 South Cass Avenue, Argonne, Illinois 60439, USA

⁸Mathematical Sciences Institute, The Australian National University, Canberra, ACT 0200, Australia

⁹School of Computer Science, University of Manchester, Oxford Road, Manchester M13 9PL, UK

Correspondence to: S. Valcke (valcke@cerfacs.fr)

Received: 29 June 2012 – Published in Geosci. Model Dev. Discuss.: 25 July 2012

Revised: 12 November 2012 – Accepted: 13 November 2012 – Published: 21 December 2012

Abstract. This paper presents a review of the software currently used in climate modelling in general and in CMIP5 in particular to couple the numerical codes representing the different components of the Earth System. The coupling technologies presented show common features, such as the ability to communicate and regrid data, and also offer different functions and implementations. Design characteristics of the different approaches are discussed as well as future challenges arising from the increasing complexity of scientific problems and computing platforms.

of the constituent models. In general, coupling data must be regridded and passed between the components subject to different constraints such as conservation of physical quantities, stability of the flux exchange numerics, consistency with physical processes occurring near the component surface, etc. In addition, computational efficiency of the coupling on parallel hardware is of course required. This paper provides a review and a short comparative analysis of the main coupling technologies currently used in Earth System Modelling.

1 Introduction

Model coupling is essential for realizing multi-physics simulations based on two or more computing applications. An Earth System Model (ESM) is a quintessential example of a coupled model, which involves several interacting components simulating the atmosphere, oceans, land, and sea ice. The software that links together these model components is called a “coupler”. Although their implementations differ vastly, couplers used in the geophysical community typically carry out similar functions such as managing data transfer between two or more components, interpolating the coupling data between different grids, and coordinating the execution

2 The Earth System Modelling Framework

The Earth System Modelling Framework (ESMF, www.earthsystemmodeling.org) is open source software for building modelling components and coupling them together to form weather, climate, coastal, and other applications. It is used and managed by a consortium of US agencies.

ESMF is comprised of a superstructure of coupling tools and component wrappers with standard interfaces, and an infrastructure of utilities for common functions, including calendar management, message logging, grid transformations, and data communications (Hill et al., 2004). Infrastructure utilities, including a tool for generation of interpolation weights, can be used independently from the superstructure.

ESMF offers two kinds of component wrappers: a Gridded Component which is associated with a physical domain, and a Coupler Component for transforming and transferring data between Gridded Components. ESMF components exchange information with other components through state objects, which contain representations of physical fields.

ESMF enables components to run sequentially, concurrently, or in mixed mode. Components can be nested. Applications usually run with all components linked into a single executable program, but there is also support for running separate components as multiple executables or Web services. ESMF can couple components written in Fortran or C, and it includes a Python interface to regridding functions. Its component wrappers may be layered on top of other coupling technologies (e.g. MCT, FMS).

Generation of interpolation weights and their application via sparse matrix multiplication are both implemented as parallel operations. ESMF supports first order conservative, bilinear, and a higher-order finite element-based patch recovery method for remapping in 2-D and in some cases 3-D. Logically rectangular and unstructured grids are both supported. There is a range of options with respect to masking and handling of poles and unmapped points. The remapping system is flexible and modular; the calculation of interpolation weights can be performed either during a model run or offline, and the application of weights can be made as a separate call.

For the most part, ESMF methods do not modify user data numerically, and thus have no effect on the bit-for-bit characteristics of the model code. However, in the case of the sparse matrix multiplication used to apply interpolation weights, user data is directly manipulated by ESMF. In order to help users with the implementation of their bit-for-bit requirements, while also considering the associated performance impact, the ESMF sparse matrix implementation provides three levels of bit-for-bit support. The strictest level ensures that the numerical results are bit-for-bit identical, even when executing across different numbers of processors. In the relaxed level, bit-for-bit reproducibility is guaranteed when running across an unchanged number of processors, while the lowest level makes no guarantees about bit-for-bit reproducibility. The lowest level provides the greatest performance potential for those cases where numerical round-off differences are acceptable.

Metadata is an important aspect of model documentation and interoperability. Methods of the ESMF Attribute class can be used to store, aggregate and output model metadata. Metadata is organised into packages, following community conventions such as the Climate and Forecast conventions (see cf-pcmdi.llnl.gov), ISO standards, and the METAFOR Common Information Model (Lawrence et al., 2012). Metadata packages can also be customised.

In order to adopt ESMF, modelers arrange their code as a set of Gridded Components and Coupler Components, and then split these components into standard ESMF methods

(initialise, run, and finalise, each of which may have multiple phases). The next steps are to wrap native model data structures with ESMF data structures, and then register components with the framework. These wrapped components can then be called by a driver, which can be user-customised, to form a coupled application.

ESMF accommodates many implementation and sequencing options but using the component wrappers alone does not guarantee that components will be interoperable. In order to increase the level of interoperability among ESMF-based systems, a collaboration led by US operational weather centres has introduced conventions and generic templates for drivers, components, mediators, and simple connectors. This “NUOPC Layer” (National Unified Operational Prediction Capability) is scheduled for public release in early 2013.

Timing results for a variety of codes show that the sparse matrix multiply and basic component wrappers scale to tens of thousands of processors. Grid remapping and parallel communications are also fast and scalable. The framework is very robust and is exhaustively tested nightly on 24+ platforms using a suite of over 4000 tests and examples. Since it is a large package that encompasses many functions and features, users can expect a significant learning curve to master the software.

ESMF tools were used in two CMIP5 models: NASA GEOS-5 and CCSM4/CESM1. The GEOS-5 model uses ESMF throughout and is structured as a deeply nested hierarchy of ESMF components. In CCSM4/CESM1, higher order interpolation weights generated by the ESMF offline tool were used to significantly reduce interpolation noise when mapping wind stress from atmosphere to ocean, relative to the bilinear method used previously. Newer versions of CESM use weights generated by the ESMF tool extensively for reasons of speed, accuracy, and ability to handle many types of grids. ESMF component interfaces are now supported in CESM as well. In all, ESMF is used for coupling in about 12 different modelling systems, totaling about 85 different components (see www.earthsystemmodeling.org/components/). ESMF regridding utilities are also being used within data analysis and visualization packages such as the NCAR Command Language (NCL) and the Ultrascale Visualization – Climate Data Analysis Tools (UV-CDAT).

ESMF aims to address broader issues of interoperability as well as the mechanics of coupling. Future plans include extending ESMF functionality, addressing next generation computing challenges, and integrating ESMF components into science gateways that catalog and integrate diverse resources.

3 The new CPL7 coupler designed for CCSM4 and CESM1

The Community Climate System Model (CCSM) development is based at the National Center for Atmospheric

Research (NCAR) in Boulder, Colorado, USA. CCSM is a state-of-the-art global climate model consisting of four fundamental physical components: an atmosphere model, an ocean model, a land surface model, and a sea ice model. In addition, a coupler (or driver) is used to exchange boundary data between the components and to coordinate the time evolution of the physical models. CCSM is used to understand the Earth's global climate system, to predict the effects of climate change, and to understand past climates. It is developed as a high performance computing application but is used on a wide variety of platforms. The Community Earth System Model (CESM) is an extension of CCSM that includes an additional land-ice component, a higher altitude atmosphere model option, land and ocean biogeochemistry capabilities, and an atmospheric chemistry model.

Prior to CCSM4, CCSM system components ran concurrently as separate executables on distinct hardware processors and a separate coupler mediated communication, performed grid interpolation, and implicitly handled time integration. With the CCSM4 release in 2010, a completely new approach to coupling climate components was taken within CCSM (Craig et al., 2012). CCSM4 is a single executable implementation that contains a top-level driver and components coupled via standard init/run/finalise interfaces. Individual components in CCSM4 can be laid out on processors in relatively arbitrary ways such that components can be run on identical or independent hardware processors. The top-level driver that runs on all processors controls the processor layout and time sequencing of the components. A separate coupler component, which can run on a subset of all the processors, still exists in the system to regrid and/or merge coupling fields and carry out other coupler functions.

Components in CCSM4 are parallelised with MPI and OpenMP. The CCSM4 driver/coupler uses Model Coupling Toolkit (MCT, see Sect. 6) datatypes and methods extensively. Mapping weights are generated offline to ensure quality and reproducibility. By default, the CESM coupling operations produce roundoff level differences when processor counts or decompositions are varied, but an optional switch enforces bit-for-bit reproducibility when desired at some performance cost. A new parallel I/O (PIO) library is being used and offers improved I/O performance particularly in the area of memory scaling.

The new implementation improves performance because of greater flexibility in laying out components on hardware processors compared to the prior concurrent-only CCSM3 system. CCSM4 can run on a single processor without MPI but is also highly memory and performance scalable for runs at high resolution. The scaling of the CCSM4 coupler has been evaluated at different resolutions and on different hardware platforms on up to 10 000 processors. FLOP intensive kernels scale linearly across all processor counts, resolutions, and machines. Memory intensive operations scale linearly at lower processor counts, but the scaling flattens out at higher processor counts as the number of gridcells per processor

decreases below a few hundred. Communication-dominated kernels tend to scale sub-linearly at lower processor counts, and scaling tends to drop off above about 1000 processors. Scaling performance for communication dominated kernels is highly dependent on the machine and resolution. Overall, the improvements in the memory and performance scaling capability of the CCSM4 coupler compared to CCSM3 are significant.

The model is now being run at global resolutions of around one tenth of a degree on tens of thousands of processors, and several thousand years worth of CMIP5 simulations have been carried out at varied resolutions and on many different hardware platforms.

4 The GFDL Flexible Modelling System

Component-based design of model codes supposes defining standard component interfaces (SCI). The Flexible Modelling System (FMS) coupler is a domain-specific SCI: it is written quite narrowly to support ESMs and recognises only a few components: an atmosphere, an ocean surface including the sea ice, a land surface, and an ocean. Any other component inherits a grid from these, e.g. atmospheric physics and chemistry from the atmosphere; terrestrial biosphere, river and land ice from the land surface; marine biogeochemistry from the ocean. In the FMS SCI, there are “slots” for each of the specific components listed above. Components must be “wrapped” in FMS-specific data structures and procedure calls.

The FMS coupler is designed to address the question of how different components of the Earth System are discretised, each one making independent discretisation choices appropriate to its particular physics. In an atmospheric model, vertical diffusion is generally treated implicitly and stability is enhanced by computing the flux at the surface implicitly along with the diffusive fluxes in the interior. Simultaneously, land or ocean surfaces with vanishingly small heat capacity should be allowed. Therefore, the vertical diffusion of temperature in a coupled atmosphere–land system may lead to a tridiagonal matrix inversion which can be solved relatively efficiently using an up–down sweep, with the particularity that some of the layers are in the atmosphere and others are in the land. Moreover, if the components are on independent grids, the key flux computation at the surface is a physical process that must be modelled on the finest possible grid. Thus, the “exchange grid” (Balaji et al., 2006) on which this computation is performed in FMS emerges as an independent component for modelling the surface boundary layer.

A grid is defined as a set of cells created by edges joining pairs of vertices defined in a discretisation. Given two grids, an exchange grid is the set of cells defined by the union of all the vertices of the two parent grids. Quantities being transferred from one parent grid to the other are first interpolated

onto the exchange grid and then averaged onto the receiving grid. The general procedure for solving the vertical diffusion is thus split into separate up and down steps. Vertically diffused quantities are partially solved in the atmosphere and then handed off to the exchange grid, where fluxes are computed. The land or ocean surface models recover the values from the exchange grid and continue the diffusion calculation and return values to the exchange grid. The computation is then completed in the up-sweep of the atmosphere. This feature is key in the design of the FMS coupler.

The FMS coupled modelling system also includes a parallel ensemble adjustment Kalman filter for data assimilation (Zhang et al., 2005) in which ensemble members are treated as concurrent components.

Since coupling involves parallel floating-point (FP) calculations, one key concern is that of reproducibility across different parallel decompositions, since FP parallel arithmetic operations can sometimes be non-associative. Ensuring bitwise reproducibility across the FMS coupler involves a cost which can be as high as 10 % on some systems. Users have the option of requiring reproducibility but this feature can be turned off if the entire run will be done without changing parallel decomposition.

The FMS coupler has been shown to scale up to O(10 000) processors with fast surface processes coupling every atmospheric time step (typically ~ 15 min) and slow processes coupling every ocean time step (typically 1 h). FMS has been active for over a decade. Its feature list and its performance still remain state-of-the-art. The versatility of FMS is seen in GFDL's approach to CMIP5. GFDL has submitted four streams of modelling results to CMIP5: these are the CM3 model (Donner et al., 2011), which includes interactive aerosol chemistry for control, historical and projection runs; two Earth System Models ESM2M and ESM2G for the carbon cycle runs (Dunne et al., 2012); high-resolution "time-slice" experiments using the HiRAM-C180 and HiRAM-C360 models (Zhao and Held, 2012); and near-term prediction experiments using a sophisticated ensemble coupled data assimilation (ECDA) system (Zhang et al., 2007) and the CM2.1 model. All of the models are built using different combinations of choices of atmospheric dynamical cores, atmospheric physics packages, ocean models and the ECDA system, all of which are available as FMS components.

5 The OASIS3 coupler

The development of the OASIS coupler started in 1991 at CERFACS. The first design focussed on flexibility (easy change of coupling parameters) and low intrusiveness (components remain almost unchanged with respect to their standalone mode). The OASIS3 coupler (Valcke, 2006, 2012) is the direct evolution of these first developments. Since 2007, OASIS3 is developed and supported thanks to an active

collaboration between CERFACS and the French Centre National de la Recherche Scientifique (CNRS).

OASIS3 is written in Fortran and C. In a coupled system assembled with OASIS3, the coupler itself forms a separate executable that performs the regridding tasks. The component models remain separate executables with main characteristics, such as internal parallelisation or I/O, untouched with respect to their uncoupled mode. To interact with the other components through the coupler, the component models need to link to the OASIS3 coupling interface library. The coupling interface library API includes calls to receive and send the coupling fields usually implemented within the model time step loop. The characteristics of the coupling exchanges, e.g. the corresponding target or source component of an exchange or the coupling frequency, are not explicitly defined in the model code but in an external configuration file written by the user. At run-time the coupling library and the coupler perform coupling exchanges according to the information contained in this file. For regridding, OASIS3 includes the SCRIP library (Jones, 1999), adding a few specific options (such as the possibility to assign the value of the nearest non-masked source neighbour to target grid points for which the original SCRIP algorithm would not assign any value at all).

For each coupling field exchange, the different parts of a coupling field sent by the source model processes are gathered by one coupler process which regrids the whole coupling field and distributes it to the target model processes. OASIS3 can therefore be parallelised on a field-per-field basis in the sense that each coupler process can treat a subset of coupling fields. But even in this case, reproducibility with respect to parallelism is not an issue as all transformations are done, for each particular coupling field, on only one coupler process.

OASIS3 success up to now can be explained by its great flexibility and its low intrusiveness in the component codes. OASIS3 is used today by about 35 different climate modelling groups in Europe, Australia, Asia and North America. A detailed list of users can be found in Valcke (2012). In particular, OASIS3 is the coupling software used in 5 of the 7 European ESMs participating to CMIP5, i.e. CNRM-CM5 (Voldoire et al., 2011), IPSL-CM5 (Dufresne et al., 2012), CMCC-ESM (Vichi et al., 2011; Scoccimarro et al., 2011), EC-Earth (Hazelger et al., 2011), and MPI-ESM from the Max Planck Institute. OASIS3 is also used successfully in a few relatively high-resolution ($\sim 2/3^\circ$) configurations but its limited parallelism will eventually become a bottleneck in the coupled simulation. Within the framework of the current EU FP7 IS-ENES (see is.enes.org) project, work continues to parallelise and extend the existing functionality and to establish comprehensive services around OASIS3 (see oasis.enes.org).

6 The Model Coupling Toolkit

The Model Coupling Toolkit, MCT (Larson et al., 2005; Jacob et al., 2005), embodies an application-neutral approach for creating coupled multi-physics models. Thus, MCT can be used in diverse scientific fields. Because MCT imposes no architecture on the application, the developer can freely choose the number of executables and model process composition. An API also allows the developer to choose such elements as coupling data description, parallel coupling field exchanges, and support for parallel data transformation and interpolation. The MCT design philosophy is that flexibility and minimal invasiveness are vital to the development of long-life-cycle coupled models.

MCT provides a Fortran-based object model for coupling construction and bindings for C++ and Python have been developed. An MCT datatype describes the coupled system processor layout. MCT stores coupling field data in an object that supports arbitrary numbers of real- and integer-valued fields, indexed using string tokens. A domain decomposition descriptor (DDD) object uses a 1-D global index space (i.e. a linearization) to represent multidimensional index spaces. Parallel communication schedules are computed automatically from source and destination DDDs. Parallel data transfer is accomplished by calling paired send/receive methods with data storage and communication schedule datatypes as inputs. MCT provides distributed storage for precomputed interpolation coefficients from which it derives communication schedules for parallel interpolation. This operation can optionally be constrained to give identical results on different numbers of processes. MCT assumes MPI-based parallelism but includes a small MPI-replacement library for non-parallel applications.

MCT is highly portable and uses a GNU autoconf-based build system. MCT programming model derives from Fortran 90: module use to access MCT classes and methods, declaration of variables of MCT datatypes, and invocation of MCT methods to perform coupling operations. To use MCT, the developer locates logical interaction points in the subsystem models, adds code to declare and initialise MCT datatypes for coupling, and inserts handshaking calls between model pairs to initialise communication schedules. Within the model run method, the user inserts calls to load the coupling data into MCT datatypes and calls MCT parallel communication and interpolation methods.

The biggest challenge in using MCT is defining linearizations of mesh and index spaces. Most new MCT users, however, quickly build their own parallel coupled models after experimenting with the examples provided. Ease of use is the primary benefit of MCT. Its limitations are lack of support for computation of interpolation weights and for MPI communicator construction. MCT has been the basis of all CCSM couplers since 2004, supporting thousands of model-years of coupled climate simulation. In particular, the MCT-based coupler, CPL7, is being used for all of

the CMIP5 integrations being performed using CCSM4 and CESM1 (see Sect. 3). MCT has also been used to form other coupled systems; a list is available on the MCT website (www.mcs.anl.gov/research/projects/mct/).

Exascale platforms will require refactoring key MCT portions. Paucity of per-core memory at exascale requires re-examination of field data copying and DDD replication. Employing compatible mesh representation software in all components could eliminate field data copying. Employing space-filling curves as compact virtual linearisations could minimise DDD replication costs. Tolerating hardware faults and dynamic load balance means the MCT assumption of a static processor pool must be revisited. Work is under way to implement coupling in the presence of dynamically-varying processor pools. Currently, MCT supports applications on hundreds of thousands of processors and is well positioned for future coupled model challenges.

7 The Bespoke Framework Generator

The Bespoke Framework Generator (Ford et al., 2006; Armstrong et al., 2009) (BFG, www.cs.manchester.ac.uk/cnc/projects/bfg) owes its development to an analysis of the Met Office future coupling requirements, the requirements of the GENIE paleoclimate coupled model (Armstrong et al., 2009, www.genie.ac.uk) and Community Integrated Assessment System (CIAS, www.tyndall.ac.uk/research/cias, Warren et al., 2008). BFG allows the user to choose the underlying coupling technology, taking coupling metadata as input and generating tailored (bespoke) wrapper code to be compiled and linked with the user's code and the chosen coupling library. Separating the coupler from the science code offers an additional layer of flexibility which can improve portability, performance and maintainability, thus future-proofing the code. BFG treats transformations (such as grid transformations) in the same way as model code. Intrinsic transformations (such as those in OASIS3) can also be supported. Reproducibility is an issue for coupling systems and, as BFG targets existing coupling technologies, BFG relies on the support for reproducibility that they provide; the wrapper code produced by BFG does not itself cause any reproducibility issues. BFG has been designed to be generic and extensible; thus, BFG may be used in application domains other than ESM (Warren et al., 2008; Delgado-Buscalioni et al., 2005).

BFG2, the current implementation, can be run directly from the BFG portal (see bfg.cs.man.ac.uk). The BFG2 model API supports models written in Fortran as a module containing subroutines or a set of subroutines, C as a set of procedures and Python as a class with a set of methods. The coupled model behaviour is specified as composition and deployment metadata in XML. At the language level, BFG2 supports passing data to and from subroutines/procedures/methods via arguments and/or in-place put/get calls. The former approach is similar to that used by

ESMF, CPL7 and FMS (see Sects. 2, 3 and 4) and the latter to MPI, CCSM3 and OASIS3 (see Sects. 3 and 5). Since models are not main programs, BFG2 is able to map models for deployment either as a single executable or as multiple executables, each containing one or more models. Coupling connections can, on first use, be initialised in a variety of ways, including from a model or a file. Coupling technologies currently supported to exchange data between models include argument passing, MPI or OASIS. The OASIS BFG implementation supports the specification of grids using an XML representation of the Gridspec (Balaji et al., 2006) as well as the use of intrinsic OASIS transformations. The work was carried out on a development version of the now withdrawn OASIS4 (Redler et al., 2010) and efforts are ongoing to extend BFG to support the most recent version of OASIS, OASIS3-MCT (Valcke, 2012).

BFG2 requires model (science) code to conform to some simple coding rules and to be described by definition metadata. Composition metadata specifies how the models (and transformations) are connected together and deployment metadata specifies how to map the models onto the available hardware and software resources. BFG2 takes the metadata as input and generates bespoke control and communication code using a Python program.

In conclusion, BFG isolates science code from the coupling infrastructure and provides a metadata-driven code generation system to provide flexibility in model composition and deployment. This flexibility allows BFG to achieve similar performance to hand-written code and provides the opportunity for finer-grained coupling than is typical today. BFG is currently used within CIAS, where it is used to couple over 20 different Integrated Assessment model configurations. BFG is not used in any CMIP5 runs but offers a potential solution for the coupling of future Earth System Models.

Some limitations of BFG2 are, in particular, that wrapper code must be regenerated whenever the metadata changes and that data partition metadata for use with parallel models is not yet supported (but will be added soon, using MCT for MPI implementations). BFG2 is being extended to support models written in a less modular way – in particular, model codes which are main programs, codes with internal control, and models where the source code is not available. In the future, BFG2 will be extended to support ESMF and CPL7 as coupler targets. In addition, the feasibility of using BFG2 to couple together models that conform to different frameworks by generating appropriate adaptor code will be investigated.

8 Conclusions

This paper provides an overview of the current coupling technologies used in Earth System Modelling. Since no quantitative information was collected, conclusions are limited to the likely outcomes of the different design strategies. While the details of the approaches vary, features of the different

coupling technologies typically include an ability to communicate data between components, regrid data, and manage the time evolution of the model integration.

Coupling using a concurrent multiple executable approach (e.g. OASIS3) requires minimal modification to existing component codes but limits the ways they can be mapped to hardware, which can hinder performance. Coupling via component-level interfaces within one integrated application (e.g. ESMF, CPL7 or FMS) generally requires users to split components into initialise, run, and finalise methods, and may limit the places where data exchanges can happen. Although this can simplify program flow, it can also affect time sequencing and require scientific reformulation; however, because components can be run sequentially or concurrently, there are additional opportunities for performance optimisation. This “integrated” approach also enables components to be nested, with multiple coupling levels. Coupling toolkits (e.g. MCT) are designed for a la carte use of classes and methods. They allow great flexibility for building custom parallel coupling mechanisms, with either single or multiple executable approaches. Subsets of other coupling technologies (e.g. ESMF utilities) can also be used separately to solve some coupling problems. Research in generative programming (e.g. BFG) explores potential ways to unify the different coupling approaches. In the end, science needs both flexible coupling capabilities and high performance. Both have become crucial in the last few years as coupling complexity and resolution have rapidly increased and these trends are expected to continue in the future. Qualitative comparison of the performances of the different approaches is an under-researched problem and there are plans in the IS-ENES2 EU project, which is a follow-on of the IS-ENES project funded over the 2013–2017 period, to develop a benchmark suite to address this issue.

Continual improvement in coupled climate model performance may become more difficult. Most of the gains in the last decade came from faster hardware on a per-processor basis and improvements in grid decompositions, memory parallelisation, and communication algorithms. Unfortunately, future generation hardware is likely to consist of orders of magnitude more processors that are slower, heterogeneous, and with less and slower memory. Moving into the exascale era will require, for coupling technology as for other software, both finding additional opportunities for parallelism and improving communication mechanisms to better overlap communication with computation.

Over much of the past two decades, several groups have worked relatively independently to develop Earth System coupling technology. In many aspects, those implementations have converged as individuals gain experience and as common science and high performance requirements drive implementations. At the same time, different scientific communities continue to benefit from fundamentally different solutions. Moving forward, the community recognises the potential benefit of much closer collaboration especially

considering the uncertainty of future hardware. In addition, there is recognition that if future hardware requires significant rewrites of Earth System Models in new programming languages, an opportunity might present itself to unify coupling approaches and share developmental costs.

Acknowledgements. This paper is based on the outcome of the Workshop on “Coupling Technologies for Earth System Modelling: Today and Tomorrow” (Valcke and Dunlap, 2011) organised in CERFACS in 2010. We would like to thank all participants who took actively part in the discussions and the EU FP7 IS-ENES project (Contract GA No: 228203) for financial support.

Edited by: D. Ham



The publication of this article is financed by CNRS-INSU. This work was supported in part by the Office of Advanced Scientific Computing Research, Office of Science, U.S. Dept. of Energy, under Contract DE-AC02-06CH11357.

References

- Armstrong, C. W., Ford, R. W., and Riley, G. D.: Coupling integrated Earth System Model components with BFG2, *Concurr. Comp-Pract. E*, 21, 767–791, doi:10.1002/cpe.1348, 2009.
- Balaji, V., Anderson, J., Held, I., Winton, M., Durachta, J., Malyshev, S., and Stouffer, R. J.: The Exchange Grid: a mechanism for data exchange between Earth System components on independent grids, in: Proceedings of the 2005 International Conference on Parallel Computational Fluid Dynamics, College Park, MD, USA, Elsevier, 2006.
- Craig, A. P., Vertenstein, M., and Jacob, R.: A New Flexible Coupler for Earth System Modeling developed for CCSM4 and CESM1, *Int. J. High Perform. C*, 26-1, 31–42, doi:10.1177/1094342011428141, 2012.
- Delgado-Buscalioni, R., Coveney, P. V., Riley, G. D., and Ford, R. W.: Hybrid Molecular-Continuum Models under the General Coupling Framework, *Philos. T. Roy. Soc. A*, 363, 1975–1985, 2005.
- Donner, L. J., Wyman, B. L., Hemler, R. S., Horowitz, L. W., Ming, Y., Zhao, M., Golaz, J. C., Ginoux, P., Lin, S. J., Schwarzkopf, M. D., Austin, J., Alaka, G., Cooke, W. F., Delworth, T. L., Freidenreich, S. M., Gordon, C. T., Griffies, S. M., Held, I. M., Hurlin, W. J., Klein, S. A., Knutson, T. R., Langenhorst, A. R., Lee, H.-C., Lin, Y., Magi, B. I., Malyshev, S. L., Milly, P. C. D., Naik, V. and Nath, M. J., Pincus, R., Ploshay, J. J., Ramaswamy, V., Seman, C. J., Shevliakova, E., Sirutis, J. J., Stern, W. F., Stouffer, R. J., Wilson, R. J., Winton, M., Wittenberg, A. T., and Zeng, F.: The dynamical core, physical parameterizations, and basic simulation characteristics of the atmospheric component AM3 of the GFDL Global Coupled Model CM3, *J. Climate*, 24, 3484–3519, 2001.
- Dufresne, J.-L., Foujols, M.-A., Denvil, S., Caubel, A., Marti, O., Aumont, O., Balkanski, Y., Bekki, S., Bellenger, H., Benshila, R., Bony, S., Bopp, L., Braconnot, P., Brockmann, P., Cadule, P., Cheruy, F., Codron, F., Cozic, A., Cugnet, D., de Noblet, N., Duvel, J.-P., Ethé, C., Fairhead, L., Fichefet, T., Flavoni, S., Friedlingstein, P., Grandpeix, J.-Y., Guez, L., Guilyardi, E., Hauglustaine, D., Hourdin, F., Idelkadi, A., Ghattas, J., Jous-saume, S., Kageyama, M., Krinner, G., Labetoulle, S., Lahellec, A., Lefebvre, M.-P., Lefevre, F., Levy, C., Li, Z. X., Lloyd, J., Lott, F., Madec, G., Mancip, M., Marchand, M., Masson, S., Meurdesoif, Y., Mignot, J., Musat, I., Parouty, S., Polcher, J., Rio, C., Schulz, M., Swingedouw, D., Szopa, S., Talandier, C., Terray, P., and Viovy, N.: Climate change projections using the IPSL-CM5 Earth System Model: from CMIP3 to CMIP5, *Clim. Dynam.*, submitted, 2012.
- Dunne, J. P., John, J. G., Adcroft, A. J., Griffies, S. M., Hallberg, R. W., Shevliakova, E. N., Stouffer, R. J., Cooke, W., Dunne, K. A., Harrison, M. J., Krasting, J. P., Levy, H., Malyshev, S. L., Milly, P. C. D., Phillipps, P. J., Sentman, L. T., Samuels, B. L., Spelman, M. J., Winton, M., Wittenberg, A. T., and Zadeh, N.: GFDL’s ESM2 global coupled climate-carbon Earth System Models Part I: Physical formulation and baseline simulation characteristics, *J. Climate*, 25, 6646–6665, 10.1175/JCLI-D-11-00560.1, 2012.
- Ford, R. W., Riley, G. D., Bane, M. K., Armstrong C. W., and Freeman, T. L.: GCF: A General Coupling Framework, *Concurr. Comp-Pract. E*, 18, 163–181, 2006.
- Hazeleger, W., Wang, X., Severijns, C., Stefanescu, S., Bintanja, R., Sterl, A., Wyser, K., Semmler, T., Yang, S., van den Hurk, B., van Noije, T., van der Linden, E., and van der Wiel, K.: EC-Earth V2.2: description and validation of a new seamless earth system prediction model, *Clim. Dynam.*, 39, 2611–2629, doi:10.1007/s00382-011-1228-5, 2011.
- Hill, C., DeLuca, C., Balaji, V., Suarez, M., and da Silva, A.: Architecture of the Earth System Modeling Framework, *Comput. Sci. Eng.*, 6, 18–28, 2004.
- Jacob, R., Larson, J., and Ong, E.: M x N Communication and Parallel Interpolation in Community Climate System Model Version 3 Using the Model Coupling Toolkit, *Int. J. High. Perform. C*, 19, 293–307, 2005.
- Jones, P.: Conservative remapping: First- and second-order conservative remapping, *Mon. Weather Rev.*, 127, 2204–2210, 1999.
- Larson, J., Jacob, R., and Ong, E.: The Model Coupling Toolkit: A New Fortran90 Toolkit for Building Multiphysics Parallel Coupled Models, *Int. J. High Perform. C*, 19, 277–292, 2005.
- Lawrence, B. N., Balaji, V., Bentley, P., Callaghan, S., DeLuca, C., Denvil, S., Devine, G., Elkington, M., Ford, R. W., Guilyardi, E., Lautenschlager, M., Morgan, M., Moine, M.-P., Murphy, S., Pascoe, C., Ramthun, H., Slavin, P., Steenman-Clark, L., Toussaint, F., Treshansky, A., and Valcke, S.: Describing Earth System Simulations with the Metafor CIM, *Geosci. Model Dev. Discuss.*, 5, 1669–1689, doi:10.5194/gmdd-5-1669-2012, 2012.
- Redler, R., Valcke, S., and Ritzdorf, H.: OASIS4 – a coupling software for next generation earth system modelling, *Geosci. Model Dev.*, 3, 87–104, doi:10.5194/gmd-3-87-2010, 2010.
- Scoccimarro, E., Gualdi, S., Bellucci, A., Sanna, A., Fogli, P. G., Manzini, E., Vichi, M., Oddo, P., and Navarra, A.: Effects of Tropical Cyclones on Ocean Heat Transport in a High Resolution Coupled General Circulation Model, *J. Climate*, 24, 4368–4384, 2011.

- Valcke, S.: OASIS3 User Guide (prism_2-5), CERFACS Technical Report TR/CMGC/06/73, CERFACS, Toulouse, France, 60 pp., 2006.
- Valcke, S.: The OASIS3 coupler: a European climate modelling community software, *Geosci. Model Dev. Discuss.*, 5, 2139–2178, doi:10.5194/gmdd-5-2139-2012, 2012.
- Valcke, S. and Dunlap, R.: Report from the workshop “Coupling Technologies for Earth System Modelling: Today and Tomorrow”, *CLIVAR Exchanges*, 16, 38–39, 2011.
- Valcke, S., Craig, T., and Coquart, L.: OASIS3-MCT User Guide(OASIS3-MCT 1.0), CERFACS Technical Report WN/CMGC/12/49, CERFACS, Toulouse, France, 46 pp., 2012.
- Vichi, M., Manzini, E., Fogli, P. G., Alessandri, A., Patara, L., Scoccimarro, E., Masina, S., and Navarra A.: Global and regional ocean carbon uptake and climate change: sensitivity to a substantial mitigation scenario, *Clim. Dynam.*, 37, 1929–1947, doi:10.1007/s00382-011-1079-0, 2011.
- Voldoire, A., Sanchez-Gomez, E., Salas y Méliá, D., Decharme, B., Cassou, C., Sénési, S., Valcke, S., Beau, I., Alias, A., Chevalier, M., Déqué, M., Deshayes, J., Douville, H., Fernandez, E., Madec, G., Maisonnave, E., Moine, M.-P., Planton, S., Saint-Martin, D., Szopa, S., Tyteca, S., Alkama, R., Belamari, S., Braun, A., Coquart, L., and Chauvin, F.: The CNRM-CM5.1 global climate model: Description and basic evaluation, *Clim Dynam.*, doi:10.1007/s00382-011-1259-y, in press, 2011.
- Warren, R., de la Nava Santos, S., Arnell, N. W., Bane, M., Barker, T., Barton, C., Ford, R., Fussel, H. M., Hankin Robin, K. S., Klein, R., Linstead, C., Kohler, J., Mitchell, T. D., Osborn, T. J., Pan, H., Raper, S. C. B., Riley, G., Schellnhuber, H. J., Winne, S., and Anderson, D.: Development and illustrative outputs of the Community Integrated Assessment System (CIAS), a multi-institutional modular integrated assessment approach for modelling climate change, *J. Environ. Model. Softw.*, 23, 592–610, doi:10.1016/j.envsoft.2007.09.002, 2008.
- Zhang, S., Harrison, M. J., Wittenberg, A. T., Rosati, A., Anderson, J. L., and Balaji, V.: Initialization of an ENSO Forecast System using a Parallelized Ensemble Filter, *Mon. Weather Rev.*, 133, 3176–3201, 2005.
- Zhang, S., Harrison, M. J., Rosati, T., and Wittenberg, A.: System Design and Evaluation of Coupled Ensemble Data Assimilation for Global Oceanic Climate Studies, *Mon. Weather Rev.*, 135, 3541–3564, 2007.
- Zhao, M. and Held, I. M.: TC-permitting GCM simulations of hurricane frequency response to sea surface temperature anomalies projected for the late-twenty-first century, *J. Climate*, 25, 2995–3009, doi:http://dx.doi.org/10.1175/JCLI-D-11-00313.1, 2012.

The submitted manuscript has been created by UChicago Argonne, LLC, Operator of Argonne National Laboratory (“Argonne”). Argonne, a U.S. Department of Energy Office of Science laboratory, is operated under Contract No. DE-AC02-06CH11357. The U.S. Government retains for itself, and others acting on its behalf, a paid-up nonexclusive, irrevocable worldwide license in said article to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, by or on behalf of the Government.